REMARKS

1. Summary of the Office Action

Claims 1-10, 12-21, 23, 26-35, 37-46 and 48 stand rejected under § 102(a) as allegedly being anticipated by U.S. Patent No. 5,557,798 to Skeen et al. (hereinafter "Skeen"). Claims 11, 22, 24-25, 36 and 47 stand rejected under § 103(a) as being obvious in view of the combination of Skeen and U.S. Patent No. 5,680,551 to Martino. Independent claims 1, 9, 12, 20, 26, 34, 37, and 45 have been amended, and claims 6, 17, 31, and 42 have been cancelled.

2. Response to § 102 Rejections

To anticipate a claim, the reference must teach every element of the claim. "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." Verdegaal Bros. v. Union Oil Co. of California, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987).

i. SKEEN DOES NOT TEACH EVERY ELEMENT OF CLAIMS 1, 12, 26 AND 37.

Claim 1 as amended, which is representative of the group, includes the following:

> registering, in response to that message, a certified message subscription request, for messages of the first type, for that subscriber application at the publisher application;

(Claim 1, as amended). In particular, the above limitation of claim 1 relates to establishing an application level certified communications session between the subscriber application and the publisher application, in part, by registering a certified message subscription request for a subscriber application at a publisher application.

An exemplary embodiment of an invention performing such an operation is described in the "Overview" section of the written description, wherein, on page 5, the second paragraph states:

> As is described in much greater detail below, a listener 20 can register with a specific publisher 10 to receive certified messages. This communication includes the subscribers name, its "inbox" address and the subject/content of messages it requires information on. Thus the publisher will have a list of subscriber names and inboxes (but know nothing else about the subscriber) for all subscribers wishing to receive certified messages.

("Overview", Page 5, Second Paragraph)

In contrast to claim 1, Skeen does not disclose registering a certified message subscription request for a subscriber application at a publisher application. Instead, according to Skeen:

> Subject-based addressing starts with a subscribe call 188 to the subject mapper 180 by a client application 16 running on host computer 10. The subscribe call is a request for information regarding a particular subject. Suppose hypothetically that the particular subject was equity.IBM.news. This subscribe call would pass two parameters to the subject mapper 180. One of these parameters would be the subject equity.IBM.news. The other parameter would be the name of a callback routine in the client application 16 to which data regarding the subject is to be passed. The subscribe call to the subject mapper 180 is a standard procedure call.

(Skeen, Col. 20, Line 23). According to Skeen, a subscriber must subscribe to receive information related to a particular subject. Skeen simply refers to a subscription request. However, the subscription request referred to in Skeen is not a certified message subscription request and does not relate to a certified communications session. Consequently, in contrast to claim 1, Skeen does not disclose registering a certified message subscription request at a publisher application.

In addition, according to Skeen, the subscription request is sent to and registered at the subject mapper, not at a publisher application, as is claimed in claim 1. Accordingly, under Skeen, in the case of a message delivery failure, the protocol engine can at best only inform the application that a subscriber did not receive some packets of a message. However, because Skeen does not involve registering a certified message subscription request at a publisher application, the publisher application has no knowledge of the subscriber application. Accordingly, under Skeen, the protocol engine has no way of notifying the publisher application which message was not received and which subscriber did not receive the message.

Claim 1, as amended, also includes the following:

> establishing, in response to the subscription request, a certified communications session between the subscriber application and the publisher application in which the publisher application communicates a subsequent message of the first type to at least the subscriber application and monitors whether the subscriber application has received the subsequent message by waiting for an acknowledgement of receipt of the subsequent message from the subscriber application and, if the acknowledgement does not arrive within a defined time, resends the unacknowledged message to the subscriber application, thereby establishing a certified message delivery session between the publisher application and the subscriber application.

(Claim 1, Emphasis added). In accordance with claim 1, a certified communication session between the subscriber application and the publisher application is established. The communication session between the publisher and the subscriber is referred to as certified, because the publisher application monitors whether the subscriber application has received a subsequent message by waiting for an acknowledgement of receipt of the subsequent message from the subscriber application and, if the acknowledgement does not arrive within a defined time, the publisher application resends the unacknowledged message to the subscriber application.

Skeen does not disclose establishing a certified communication session between a subscriber application and a publisher application. Skeen simply refers to establishing a communication session between a publisher application and a subscriber application. However, Skeen does not disclose establishing a certified communication session.

Moreover, Skeen does not disclose a publisher application monitoring whether a subscriber application has received a subsequent message by waiting for an acknowledgement of receipt of the subsequent message from the subscriber application and, if the acknowledgement does not arrive within a defined time, the publisher application resends the unacknowledged message to the subscriber application. Instead, Skeen discloses:

> One of these value added services is the reliable broadcast protocol. This protocol engine adds sequence numbers to packets of packetized messages on the transmit side and verifies that all packets have been received on the receive side. Packets are stored for retransmission on the transmit side. On the receive side, if all packets did not come in or some are garbled, a request is sent for retransmission. The bad or missing packets are then resent. When all packets have been successfully received, an acknowledgment message is sent. This causes the transmit side protocol engine to flush the packets out of the retransmit buffer to make room for packets of the next message.

(Skeen, Col. 5, Line 47, Emphasis added). Skeen discloses a reliable broadcast protocol for transmitting packets, not a publisher application monitoring whether a subscriber application has received a subsequent message. According to Skeen, at the protocol level, each message is broken down into packets to be transmitted across a network. The packets, which are subcomponents of a message and not equivalent to a message as the Examiner has suggested, are given a sequence number and sent across the network. If a packet does not arrive, or arrives out of order, the protocol engine is informed.

First, the protocol engine referred to in Skeen is not part of the publisher application, but instead, it is part of a distributed communications layer. (See Col. 5, Lines 30-45; See also, Fig. 17, clearly illustrating the distinction between the distributed communication layer 352 and the publisher/subscriber applications 340, 346 and 354). Therefore, at best, Skeen discloses a protocol engine monitoring packets, not messages. Skeen does not disclose a publisher application monitoring whether a subscriber application has received messages, as claimed in claim 1.

Furthermore, the passage in Skeen relied upon by the Examiner relates to packets, not messages. One skilled in the art will immediately recognize and understand this significant difference. For example, as stated in Skeen, sequence numbers are added to packets of packetized messages. However, packet order is insignificant as it relates to the invention as claimed in claim 1. Claim 1 refers to the monitoring of messages, not packets. For example, under Skeen, the protocol engine can at best inform a publisher application that a packet of a message was not successfully delivered. However, under Skeen, the publisher application cannot monitor whether a subscriber application has received a subsequent message. Consequently, Skeen does not disclose a publisher application monitoring whether a subscriber application has received a subsequent message.

Finally, the reliable broadcast protocol described in Skeen is significantly different from the method of claim 1. According to Skeen:

> After determining which packets are missing or garbled, if any, the receiving protocol engine then sends a message back to the communication layer of the service or publishing process. This message will either acknowledge that all packets have been received without a problem or will request that certain packets be retransmitted.

(Skeen, Col. 46, Line 18). According to Skeen, the protocol engine on the publishing side retransmits garbled or missing packets, only in response to a message requesting retransmission of the packets. For example, according to Skeen, retransmission of packets is dependent upon receiving a request for retransmission. In contrast to claim 1, Skeen does not disclose monitoring by waiting for an acknowledgement of receipt of the subsequent message from the subscriber application and, if the acknowledgement does not arrive within a defined time, the publisher application resends the unacknowledged message to the subscriber application.

ii. SKEEN DOES NOT TEACH EVERY ELEMENT OF CLAIM 23.

Claim 23, includes the following:

> labeling the outgoing message with a label including the
> delivery session name and a sequence number

As discussed above with reference to claims 1 and 12, Skeen does not disclose labeling an outgoing message with a sequence number. Skeen discloses a reliable broadcast protocol for transmitting packets. According to Skeen, at the protocol level, each message is broken down into packets to be transmitted across a network. The packets, which are subcomponents of a message and not equivalent to a message, as the Examiner has suggested, are given a sequence number and sent across the network. If a packet does not arrive, or arrives out of order, the protocol engine is informed and the packet can be retransmitted. However, labeling packets will not ensure that messages arrive in a particular order. Claim 23 refers to labeling messages, not packets. Consequently, Skeen does not disclose labeling an outgoing message with a sequence number.

In light of the above, Applicants respectfully submit that the rejection under 35 U.S.C. § 102 has been also been overcome, and withdrawal of this rejection is therefore respectfully requested.

3. Response to § 103 Rejections

To establish a **prima facie** case of **obviousness**, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure. In re Vaeck, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

i. The Prior Art References Do Not Teach or Suggest All Claim Limitations, When Considered Singularly or In Combination

Claims 11, 22, 36 and 47 depend from claims 1, 12, 26 and 37 respectively. Therefore, in order to render dependent claims 11, 22, 36 and 47 unpatentable, the combination of Skeen and Martino must teach or suggest each and every limitation of claims 11, 22, 26, and 37 including the limitation of independent claims 1, 12, 26 and 37 referred to above. However, like Skeen, Martino fails to disclose a publisher application monitoring whether the subscriber application has received the subsequent message by waiting for an acknowledgement of receipt of the subsequent message from the subscriber application and, if the acknowledgement does not arrive within a defined time, the publisher application resends the unacknowledged message to the subscriber application, as claimed in independent claims 1, 12, 26 and 37. Consequently, claims 11, 22, 26, and 37 are not rendered unpatentable under Skeen in further view of Martino.

Similarly, claims 24 and 25 are dependent on claim 23. In order to render dependent claims 24 and 25 unpatentable, the combination of Skeen and Martino must teach or suggest each and every limitation of claims 24 and 25, including the limitation of independent claim 23 referred to above. However, like Skeen, Martino fails to disclose labeling the outgoing message with a label including the delivery session name and a sequence number, as claimed in independent claim 23. Consequently, claims 24 and 25 are not rendered unpatentable under Skeen in further view of Martino.

In view of the above, it is submitted that the combination of Skeen and Martino does not disclose all the limitations of claims 1, 12, 23, 26 and 37 and, accordingly, claims 1, 12, 23, 26 and 37 are allowable. As claims 2-11, 13-22, 27-36 and 38-48 depend directly or indirectly upon independent claims 1, 12, 23, 26 and 37 respectively, they are also allowable.

In light of the above, Applicants respectfully submit that the rejection under 35 U.S.C. § 103 has been overcome, and withdrawal of this rejection is therefore respectfully requested.

4. Conclusion

Having tendered the above remarks and amended the claims as indicated herein, Applicants respectfully submit that all rejections have been addressed and that the claims are now in a condition for allowance, which is earnestly solicited.

If there are any additional charges, please charge Deposit Account No. 02-2666. If a telephone interview would in any way expedite the prosecution of the present application, the Examiner is invited to contact André Marais at (408) 947-8200 ext. 204.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: 3/8, 2004

Nathan P. Elder
Nathan P. Elder
Reg. No. P-55,150

12400 Wilshire Blvd.
Seventh Floor
Los Angeles, CA 90025-1026
(408) 947-8200